ENHANCING VIDEO RETRIVAL THROUGH BAG-OF-FEATURES AND MACHINE LEARNING FUSION

Mr. E. Ramesh
Assistant Professor in Department of Electronics Engineering, Madnapalle Institute of Technology & Science
G. Jeethendra Sairam
Department of Electronics Engineering, Madnapalle Institute of Technology & Science, Angallu, India
Iyingkaran
Department of Electronics Engineering, Madnapalle Institute of Technology & Science, Angallu, India
V Harsha Vardhan Sai
Department of Electronics Engineering, Madnapalle Institute of Technology & Science, Angallu, India

Abstract - Content-based retrieval involves searching for information based on its content rather than qualities. The issue of content-based video retrieval (CBVR) is designing systems that can accurately and automatically process massive amounts of diverse movies. In addition, a content-based video retrieval system requires initial frame extraction. Features are derived from video frames. Finally, select an efficient similarity/classifier metric and machine learning algorithm to obtain video results linked to your query. Video frames are classified using the Random Forest Classifier, a machine learning technique, after extracting Histogram of Oriented Gradients (HOG) characteristics.
Keywords: Features derived from video frames, Content Based Video Retrieval (CBVR), Histogram of Oriented Gradients, Machine Learning, Random Forest Classifier.

1 INTRODUCTION

Large digital libraries of visual content are accessed by modern information retrieval systems using picture and video search. Image retrieval algorithms have traditionally relied on metadata, including captions, keywords, titles, and descriptions. Annotation-based metadata improves search efficiency [1].

Manual annotation techniques are time-consuming, difficult, and costly. To solve these issues, researchers have focused on automatic picture annotation.

The rise of social online apps and the semantic web has led to the development of web-based picture annotation tools, providing scalable solutions to streamline annotation processes [2]. Video search is a specialized data search for finding videos in digital archives. Users can search for videos using keywords, video files/links, or by just clicking on them. The system returns videos that are "similar" to the query. Similarity between videos can be identified using Meta tags, color distribution, and region/shape features [3]. Numerous video search algorithms and methodologies have emerged. Videos Meta search detects videos using information, such as text or keywords and metadata. Content based video retrieval (CBVR) examines video content using computer vision rather than textual descriptions (traditional methods). CBVR uses content similarities (e.g. textures, colors, shapes) to find movies that match user queries or specified criteria. List of CBVR Engines: These specialized search engines find videos based on visual content characteristics like color, texture, and shape/object features. The video collection exploration paradigm uses creative approaches to successfully search and browse huge digital collections [4].

This study examines image retrieval and video search techniques, including fast search, annotation, summarization, visualization, and interaction with digital collections. Our analysis of classic and emerging methodologies aims to improve information access and retrieval in visual domains.

- **Videos Meta search**: This method looks for videos based on linked metadata, such as keywords or text.
- **Content-based video retrieval (CBVR)**: Unlike traditional methods reliant on textual descriptions, CBVR leverages computer vision to analyze the content of videos. By identifying similarities in content, such as textures, colors, or shapes, CBVR retrieves videos that match user-supplied query videos or specified video features.

- **List of CBVR Engines**: These are specialized search engines that look for videos based on visual content qualities including color, texture, and shape/object properties.
- **Video collection exploration**: This paradigm explores novel methods for searching videos, employing innovative exploration paradigms to navigate large digital repositories effectively [4].

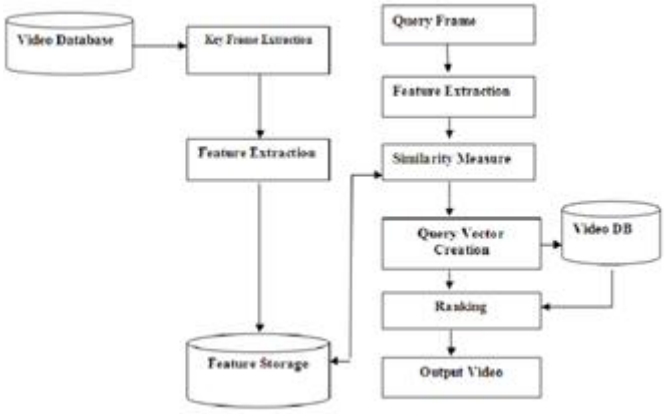

Fig. 1 General Video Retrieval System

In this paper, we delve into the intricacies of image retrieval and video search, exploring methodologies for efficient search, annotation, summarization, visualization, and interaction within digital image and video collections. We analyze both traditional approaches and emerging paradigms, aiming to contribute to the advancement of information access and retrieval in visual domains.

2 LITERATURE REVIEW

Roger Weber and Michael Mlivoncic: An RBIR system divides images into a variable number of regions and extracts a set of features from each. Next, a dissimilarity function calculates the distance between a database image and a set of reference regions. The high evaluation costs of the dissimilarity function limit RBIR to very small databases.

R. Vijaya Arjunan, Dr. V. Vijaya Kumar: Color-based searches are the most efficient and straightforward in content-based image retrieval (CBIR). However, these methods can be enhanced by incorporating some pre-processing processes. Here, the preparation techniques and picture categorization are examined. CBIR image categorization requires computing speed and efficiency. The primary advantage of this strategy is the rapid production and comparison of the applied feature vectors.

C. H. Wu and Y. J. Chen: In telephone speech recognition, the acoustic mismatch between training and testing contexts frequently results in a significant decrease in recognition ability. This work describes a keyword-driven two-level codebook-based stochastic matching (CBSM) algorithm that eliminates acoustic mismatches. Furthermore, in Mandarin speaking, recognizing the unvoiced portion of a syllable is difficult. To reduce the inaccuracy in recognizing unvoiced segments, a fuzzy search approach is presented to extract keyword candidates from a syllable lattice. Finally, a keyword relation and a weighting mechanism for keyword combinations are introduced to aid in multi-keyword detection. The core recognition units for multi-keyword detection in Mandarin speech are 94 correct context-dependent and 38 context-independent sub syllables. Each sub-syllable has a corresponding anti-sub syllable model, which is trained and verified. In this technique, 2583 professor names and 39 department names are chosen as primary and secondary keywords, respectively [6].

K. Sakthidasan alias Sankaran & V. Nagarajan: The internal and exterior data cubes are intelligently set to produce similar patches from the corresponding noise-contaminated and web photos. In this regard, two phases are used, each applying a different filtering mechanism to reduce noise. In the first step, a graph-based optimization technique is used to effectively improve patch harmonization in external denoising. In the second stage, the noise is greatly reduced by deleting the relevant internal and external

cubes. The Discrete Wavelet Transform (DWT) filtering method is used to achieve superior precision in picture denoising when compared to conventional filters[8].

Ciprian Chelba, Timothy J Hazen, Murat Saraclar: The most active sector is text-based search, which includes applications such as Web and local network searches as well as searches for personal information on one's hard drive. Speech search has received less attention, maybe because massive collections of spoken material had historically not been available. However, as storage costs have decreased and broadband access has expanded, so has the availability of online spoken audio content such as news broadcasts, podcasts, and university lectures. A number of personal and business applications exist (for example, indexing customer service calls). As data availability grows, the lack of suitable technology for processing spoken documents becomes a barrier to widespread access to spoken content [9].

Pradipta Biswas, Peter Robinson: Although user modeling is commonly utilized in HCI, there are very few systematic HCI modelling tools available for people with impairments. We are creating user models to assist with the design and evaluation of interfaces for people with diverse abilities. We provide a perceptual model that works for both visually impaired and able-bodied people. The model accepts a list of mouse events, a sequence of bitmap pictures of an interface, and the positions of various elements in the interface as input and returns a sequence of eye movements as output. Our approach accurately predicts the visual search duration for two different visual search tasks in both able-bodied and visually impaired individuals [11].

3 EXISTING METHOD

Content-Based Image Retrieval (CBIR):

Content-Based Image Retrieval (CBIR) is employed for automatic image retrieval from large databases, aiming to find images that match a query image. This technique uses visual content such as shape, color and texture to match the query image to the database photos. Three primary techniques are involved in CBIR: image retrieval by color, texture, and shape

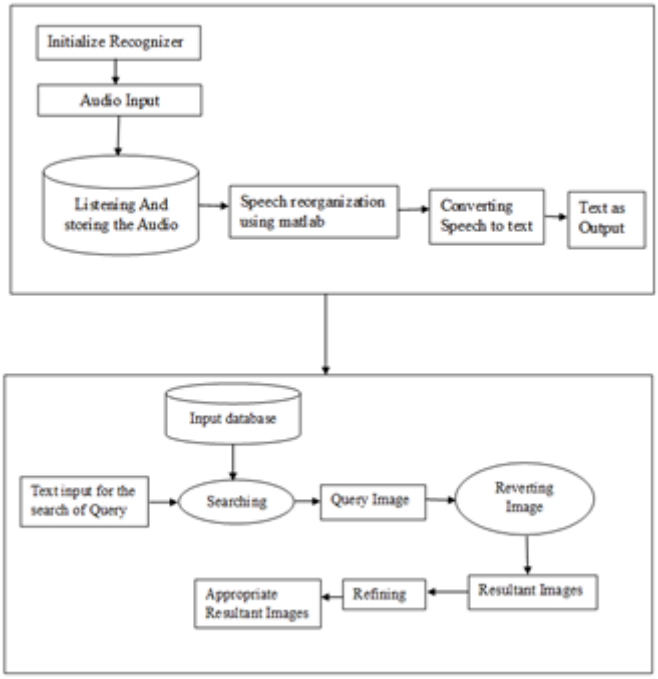

Fig. 2 Block diagram of Existing method

Techniques for CBIR:

- **Query by Example (QBE):** Users provide an example image for the system to base its search on, removing the need for textual descriptions.
- **Semantic Retrieval:** Users make requests using natural language, posing challenges due to the open-ended nature of the queries.

- **Relevance Feedback:** Users interact with the system by marking images as relevant or not relevant, refining search results iteratively.
- **Iterative/Machine Learning:** CBIR systems may incorporate machine learning techniques to improve search accuracy and relevance.
- **Other Query Methods:** Various methods, including browsing, hierarchical categories, image region queries, visual sketches, direct feature specification, and multimodal queries, enhance the flexibility of image retrieval.

Content Comparison using Image Distance Measures:
Image distance measures are used to compare images based on their content similarity. These measures consider dimensions such as color, texture, shape, etc., to compute the similarity between images. Color histograms, texture texels, and shape descriptors are common methods for representing and comparing image content

4 METHODOLOGY

This section describes our methods for developing a content-based video retrieval system. Our approach is based on two primary components: Histogram of Oriented Gradients (HOG) for visual feature extraction and Random Forest for video classification. We encode important visual information for further analysis by obtaining video data from web platforms and extracting HOG characteristics from video frames. The Random Forest method is then used to identify video frames based on these attributes, allowing for accurate content-based retrieval. We describe each step of our methodology, from data collection and feature extraction to model training and evaluation, providing a thorough picture of our system's development process.

- **Keep Data Collection:** Video data is sourced from online platforms like YouTube, forming a comprehensive database of digital videos for retrieval.
- **Feature Extraction:** Utilizing the HOG feature descriptor, features are extracted from video frames. HOG focuses on object structure and shape, providing gradient and orientation information. Firstly we will take one frame from video and that frame is converted into greyscale. Converting a color image to grayscale entails converting each pixel's color values (which commonly represent red, green, and blue channels) into a single intensity level. There are several methods for achieving this conversion, but one popular approach is to employ the luminance or average method. Here's a simple description of how it works:

1. The Luminance Method weights each pixel's color values (R, G, and B) depending on perceived intensity by the human eye. The algorithm for calculating brightness from RGB values is commonly provided as follows:

$$Y = 0.299 * R + 0.567 * G + 0.114 * B \qquad (1)$$

   - The coefficients 0.299, 0.587, and 0.114 indicate perceived brightness for the red, green, and blue channels, respectively.

2. Average Method approach involves taking the average of the RGB values:

$$Y = (R + G + B) \div 3 \qquad (2)$$

This approach averages all color channels to generate grayscale intensity.

Then it does the following steps for feature extraction:

1. **Gradiant Calculation:** Gradients are calculated using Sobel filters or other methods. Assume: G_x,G_Y Using these filters, we convolve them with the image to obtain gradients in the and directions.
2. **Gradient Orientation Binning:** Calculate gradient orientations θ for each pixel with the formula:

$$\theta = \arctan(\frac{G_Y}{G_x}) \qquad (3)$$

- For example, at pixel (1, 1), use if G_x, = 10 and G_Y = 5 then θ = arctan (5/10) = arctan (0.5) =26.5° Orientations are divided into preset bins (e.g., 9 bins ranging from 0 to 180 degrees).

3. **Histogram Calculation:** Histogram calculation involves creating cells that cover a small piece of the image. For the purposes of this example, assume that each cell is 2x2 pixels.
   - Gradient magnitudes are accumulated within each cell and binned by direction. For example, in cell (1,1), orientations 20°, 30°, 40°, and 50° with magnitudes 5, 7, 3, and 6 may result in the following histogram:

Classification with Random Forest: Random forest is a reliable and efficient machine learning technique commonly used for classification jobs. During training, the algorithm generates many decision trees and outputs either the mean prediction (regression) or the mode of the classes (classification). Random forest algorithms can handle huge datasets with high dimensionality while reducing over fitting through random feature selection and ensemble learning. Random forest can help pick and analyze features by providing insights into their relevance. Random Forest, which can handle both category and numerical data, has applications in several sectors, including healthcare. Gini Impurity:

$$Gini(D) = 1 - \sum_{i=1}^{c}(p_i)^2 \qquad (4)$$

where D is a dataset, c is the number of classes, and pi is the probability that an instance belongs to class i in dataset D. Gini impurity is a measure of dataset impurity, with lower values suggesting purer and more homogeneous subgroups.

Entropy:

$$Entropy(D) = -\sum_{i=1}^{c} p_i log_2(p_i) \qquad (5)$$

Entropy, like Gini impurity, assesses a dataset's impurity. It determines the level of ambiguity or disorder in the dataset. Entropy is a regularly used criterion for decision tree algorithms, such as random forest, to find the optimal split at each node.

Information Gain:

$$IG(D, A) = Entropy(D) - \sum_{v \in Values(A)} \frac{|D_v|}{|D|} Entropy(D_v) \quad (6)$$

where D is the dataset, A is an attribute, Values(A) are the potential values of A, and D_v is the subset of D that has attribute A with value v. Splitting a dataset based on a specific attribute reduces entropy and impurities, as measured by information gain. Random forest uses information gain to select the optimal feature for splitting at each node of the decision tree.

- **Training and Validation:** The classifier is trained on labelled data consisting of video frames with corresponding class labels (e.g., Parkinson's disease or Healthy). Validation data is utilized for fine-tuning the model and adjusting parameters to enhance classification accuracy. Let us see the comprehensive overview of the process.
  1. **Data Preparation**: Use Histogram of Oriented Gradients (HOG) to extract features from video frames, including object structure and shape. Label extracted features depending on video frame content, such as "Parkinson's disease" or "Healthy" .
  2. **Splitting data:** The labelled data is divided into training and validation sets. Training data will train the random forest classifier, while validation data will evaluate and fine-tune hyper parameters.
  3. **Train the Random Forest Classifier:** The Random Forest Classifier use training data to teach the random forest classifier. Use the Tree Bagger function or comparable methods to create a decision tree ensemble. Set up settings like the classification or regression method (str_method) and the number of trees (iNum Bags). Each decision tree is trained using a bootstrap sample of training data, with

randomly chosen feature subsets at each split, in order to add diversity. Examining the Random Forest's behavior with respect to classifier and regression.

Classification: Each sample's feature values are transmitted down each decision tree in the ensemble. Each leaf node of the tree assigns a class label based on the majority of training samples that reached that node during construction. After all trees have classified the data, their predictions (class labels) are counted. The sample's final prediction is based on majority vote. The class label with the greatest count among all trees' predictions is picked. The procedure for regression tasks is similar, but instead of class labels, each decision tree predicts a continuous value, such as a numerical score or value.

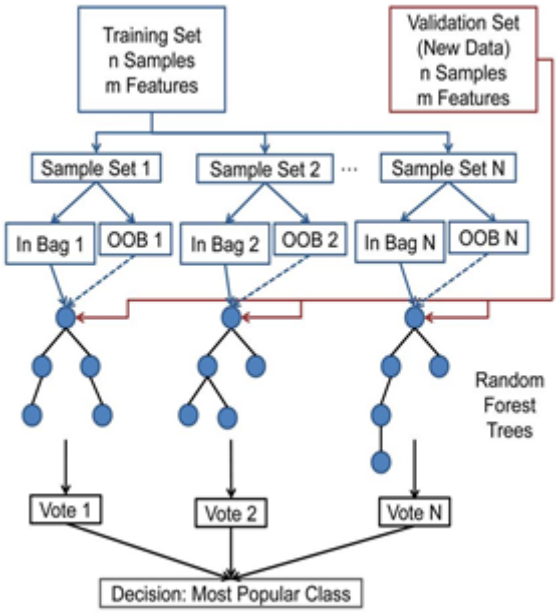

Fig. 3: Random Forest Diagram[13]

- **Testing:** Real-world video data is used to evaluate the system's performance in retrieving relevant videos. Testing data is distinct from training and validation sets to assess the model's generalization ability.
- **Content-Based Video Retrieval System:** The developed system enables users to query videos based on content attributes such as colors, shapes, and textures, without relying on metadata. Visual content analysis facilitates retrieval of relevant matches from the database.
- **Evaluation Metrics:** Performance evaluation assesses the effectiveness of the suggested approach using criteria such as precision and recall. Precision represents result correctness, whereas recall evaluates retrieval completeness. Let us see the some commonly used evaluation metrics:

- Accuracy evaluates what portion of the total instances is classified correctly.

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions} \quad (7)$$

- Precision checks how many retrieved instances are pertinent to the subject, i.e., it measures the proportion of true positive in all positive predictions.

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \quad (8)$$

- Recall (Sensitivity) assesses if we have been able to retrieve all relevant instances or not which is measured as proportion of true positives out of actual positives.

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives} \quad (9)$$

- F1 Score – The F1 score is a type of metric that uses both precision and recall into a single value, expressing their harmonic mean. This statistic provides a balanced evaluation of a classifier's performance in classification tasks.

$$F1\ Score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (10)$$

5 PROPOSED SYSTEM

Our proposed method for content-based video retrieval consists of two steps. First, we collect and process photos of cats and woods to extract HOG features and train a Random Forest model. Then, we process videos frame by frame, predict labels using the trained model, and return suitable films based on user-provided keywords. This approach allows for rapid and accurate video retrieval, improving user experience and accessibility.

The whole recommended technique will be done in two phases that is **Training the Random Forest**, **Predicting Labels for Video in User Database and Retrieving the Video**. Let us see the detailed steps behind the each phases.

A Training the Random Forest:

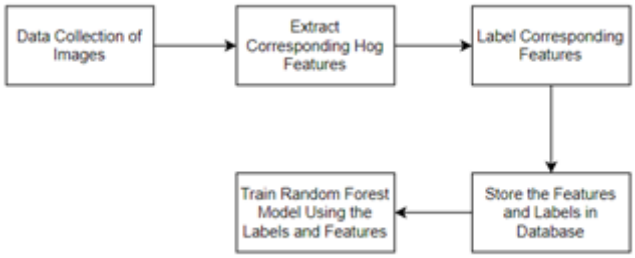

Fig. 4 Training the Random Forest

- **Step 1:** Images of cats and forests are collected from online sources and stored in separate folders named "cat" and "forest" respectively.
- **Step 2:** The code loops through each subfolder, reads images, and extracts Histogram of Oriented Gradient (HOG) features while labeling them according to their respective subfolder names.
- **Step 3:** HOG features are calculated for each image in every subfolder, and these features are stored in a 'features.mat' file along with labels.
- **Step 4:** The HOG features and labels from the 'features.mat' file are used to train a random forest model. The trained model is saved for future use.

B Predicting Labels for Videos in User Database

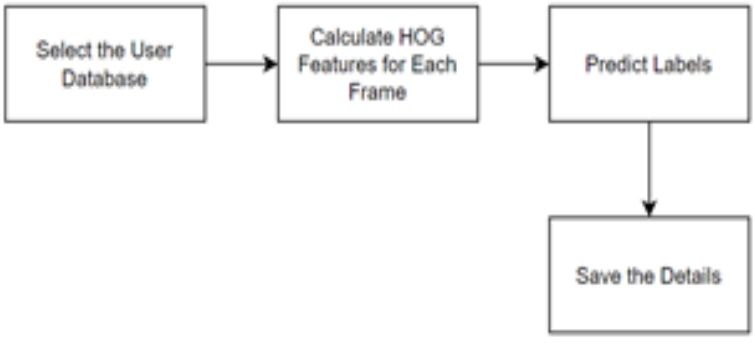

Fig. 5 Predicting Labels for Videos in User Database

- **Step1:** The user selects the user database folder containing video files.
- **Step 2:** Each video is read frame by frame, and HOG features are calculated for each frame.
- **Step 3:** These features are then used to predict labels using the trained random forest model.

C. Retrieving the Video:

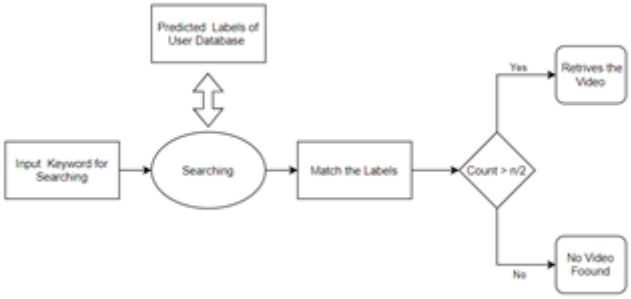

Fig. 6 Retrieving the video

- **Step1:** The user enters a keyword for searching the video.
- **Step 2:** Search keyword occurrences within the predicted labels.
- **Step 3:** Store video names where the count exceeds half the size of predictions array
- **Step 4:** Display the resultant videos.

6 RESULTS

The provided method efficiently trains a random forest model on photos of cats and woods from a user-specified dataset. It then successfully predicts labels for video frames inside a user-provided video dataset based on a search query, using the trained model. The system demonstrates a seamless combination of image processing and machine learning algorithms for effective video content analysis and retrieval.

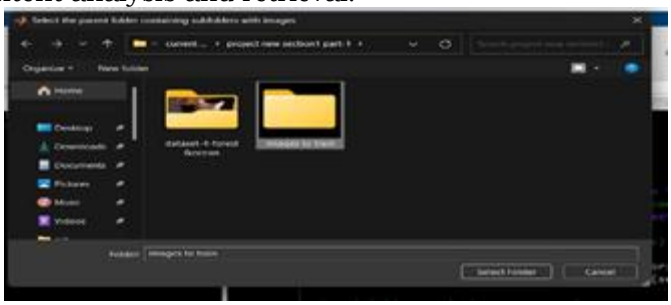

Fig. 7 Selection of images for training

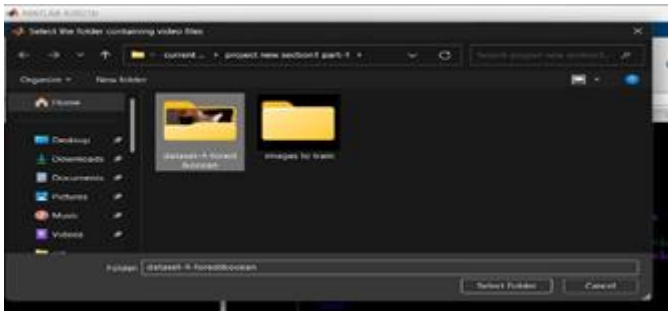

Fig. 8 Selection of user database counting videos

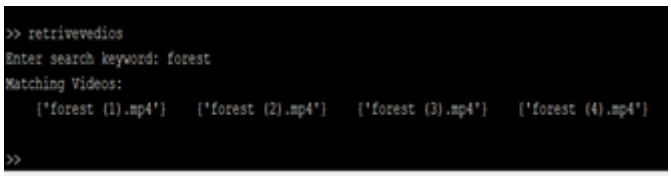

Fig. 9 Enter the keyword for searching

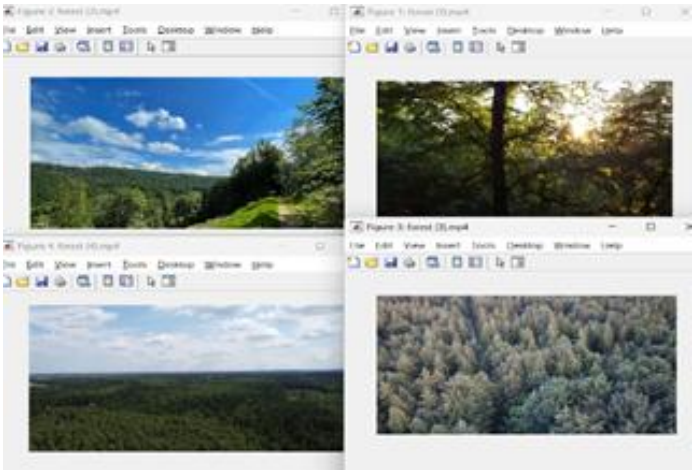

Fig. 10 Displaying the results

7 PERFORMANCE ANALYSIS

A. Performance analysis of Random Forest:

The below table describes the performance metrics of the classification model trained on the dataset. The model's effectiveness in binary classification tasks is measured using several key measures, including accuracy, precision, recall, and F1 score. These metrics provide useful information about the model's capacity to correctly categorize examples and its overall effectiveness in discriminating between positive and negative classes.

Table 1 Performance Metrics of the Classification Model

Metrics	Value
Accuracy	91.67%
Precision	100%
Recall	83.33%
F1 score	90.01%

The below confusion matrix visualizes the model's classification performance by depicting the counts of true positives, true negatives, false positives, and false negatives. This image gives a clear picture of the model's capacity to correctly categorize cases and detect classification errors.

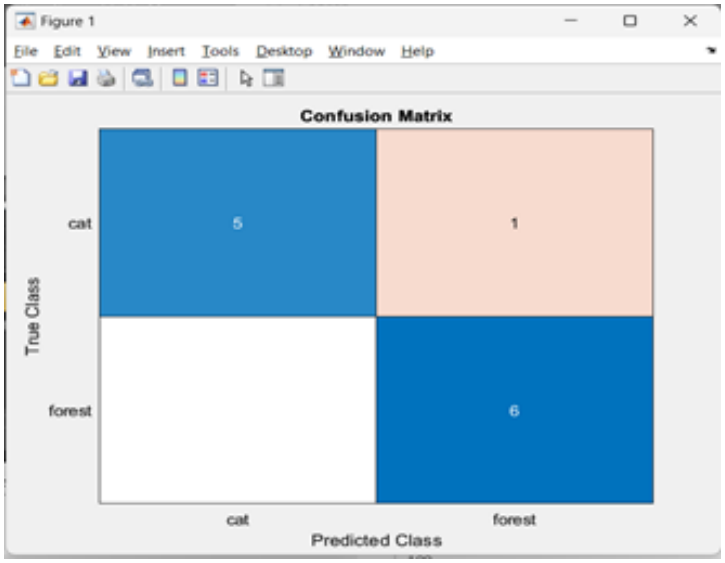

Fig. 11 Confusion matrix

B. Performance analysis of the project:

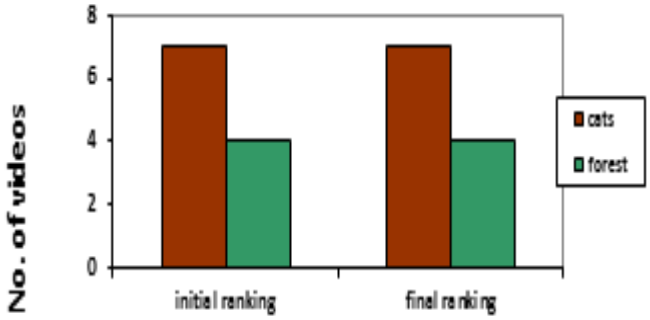

Fig. 12 Bar graph result

The bar graph above provides insights into the database's composition by displaying the distribution of videos categorized as 'cat' and 'forest.' According to the initial ranking, the database includes seven 'cat' videos and four 'forest' videos. We successfully retrieved all 'cat' videos and 'forest' videos, which is indicated by the final ranking on the same graph. This demonstrates how accurate and effective our solution is at obtaining suitable videos based on user-entered keywords.

8 CONCLUSION

This study created a robust video retrieval system employing Histogram of Oriented Gradient (HOG) features in the MATLAB environment. The system accurately predicts labels for frames in user-provided video files after training a random forest classifier on a dataset of images. Using a search feature based on user-specified keywords, the system correctly detects and gets relevant videos from the database, demonstrating its use for content-based video retrieval tasks.

REFERENCES

1. D. A. James, “A system for unrestricted topic retrieval from radio news broadcasts,” in Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing, pp. 279–282,1994.
2. B. Logan, J.-M. Van Thong, and P. J. Moreno, “Approaches to reduce the effects of OOV queries on indexed spoken audio,” IEEE Trans. Multimedia, vol. 7, no. 5, pp. 899–906, Oct. 2005.
3. K. Sakthidasan @ Sankaran, S. Bhuvaneshwari and Dr. V. Nagarajan “A new edge preserved technique using iterative median filter” in IEEE International Conference on Communication and Signal Processing(ICCSP 2014), pp:1750 – 1754, 2014.
4. JIEEE Trans. Acoust., Speech, Signal Process, vol. ASSP-33, no. 3, pp. 587–594, 1985. “A modified K-means clustering technique for application in isolated work recognition” was developed by J. G. Wilpon and L. R. Rabiner.
5. K. Sakthidasan @ Sankaran, G. Ammu and Dr. V. Nagarajan “Non local video restoration using iterative method” in IEEE International Conference on Communication and Signal Processing(ICCSP 2014), pp:1740-1744,2014.
6. C.-H. Wu and Y.-J. Chen, “Multi-keyword spotting of telephone speech using a fuzzy search algorithm and keyword-driven two-level CBSM,” Speech Commun., vol. 33, pp. 197– 212, 2001.
7. B. Chen, H.-M. Wang, and L.-S. Lee, “Discriminating capabilities of syllable-based features and approaches of utilizing them for voice retrieval of speech information in Mandarin Chinese,” IEEE Trans. Speech Audio Process., vol. 10, no. 5, pp. 303–314, Jul. 2002.
8. K. Sakthidasan alias Sankaran & V. Nagarajan, “Noise Removal through the Exploration of Subjective and Apparent Denoised Patches Using Discrete Wavelet Transform”, IETE Journal of Research, ISSN: 0377-2063, 2019, DOI: 10.1080/03772063.2019.1569483
9. Ciprian Chelba, Timothy J Hazen, Murat Saraclar, “Retrieval and browsing of spoken content”, in IEEE Signal Processing Magazine, vol. 25, no. 3, pp: 39-49, 2008.
10. Jonathan Mamou, Bhuvana Ramabhadran, “Phonetic query expansion for spoken document retrieval”, in 9th Annual Conference of the International Speech Communication Association, Brisbane, Australia, September 22-26, 2008.
11. Pradipta Biswas, Peter Robinson, “Modelling perception using video processing algorithms”, in Proceedings of the 23rd British HCI Group Annual Conference on People and Computers: Celebrating People and Technology, pp: 494-503, 2009.
12. Yuk Wah Wong, Dominic Widdows, Tom Lokovic, Kamal Nigam , “Scalable attribute-value extraction from semi-structured text”, in IEEE International Conference on Data Mining Workshops, pp.302- 307, 2009.
13. GitHub - learn-co-curriculum/dsc-random-forests.